Pandia: comprehensive contention-sensitive thread placement

Daniel Goodman
Georgios Varisteas
Tim Harris

daniel.goodman@oracle.com

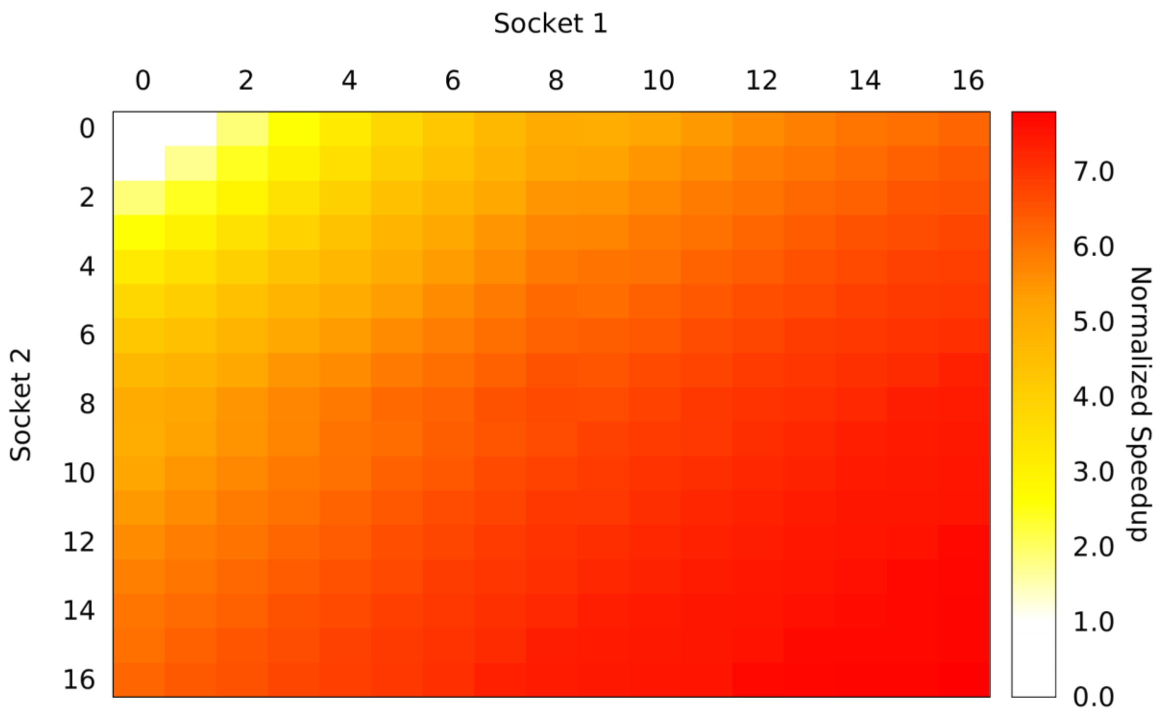
ORACLE

Safe Harbor Statement

The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle's products remains at the sole discretion of Oracle.

Example behaviour patterns

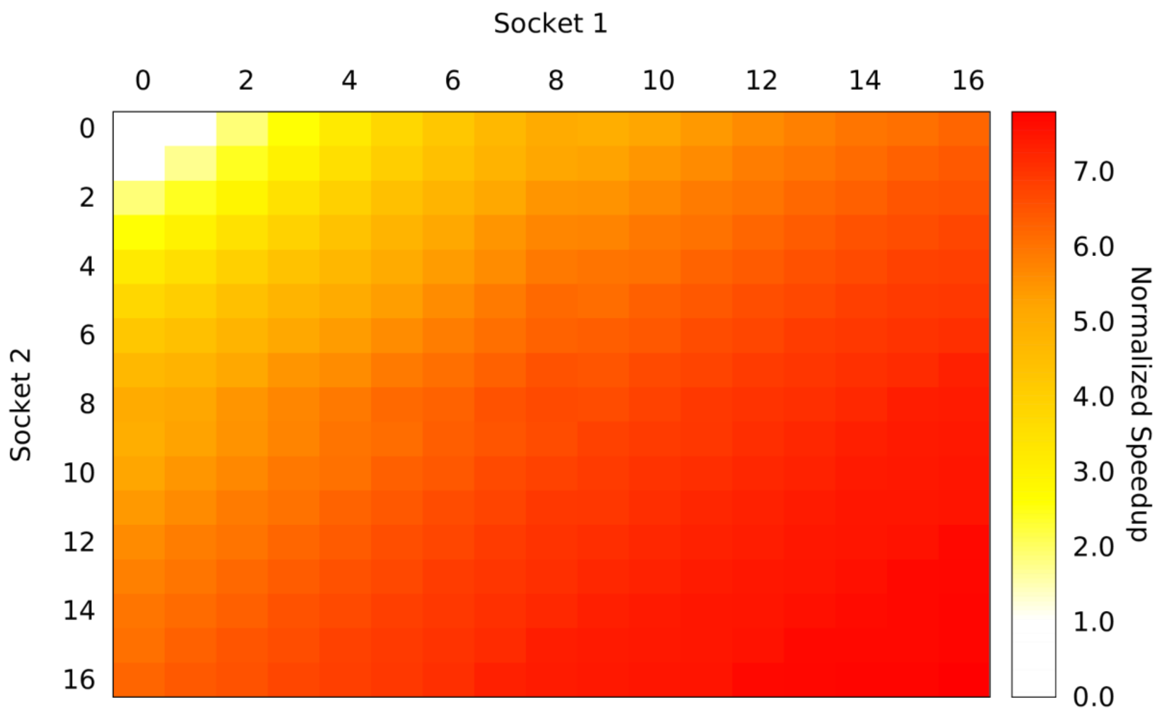
Parallel radix join optimized



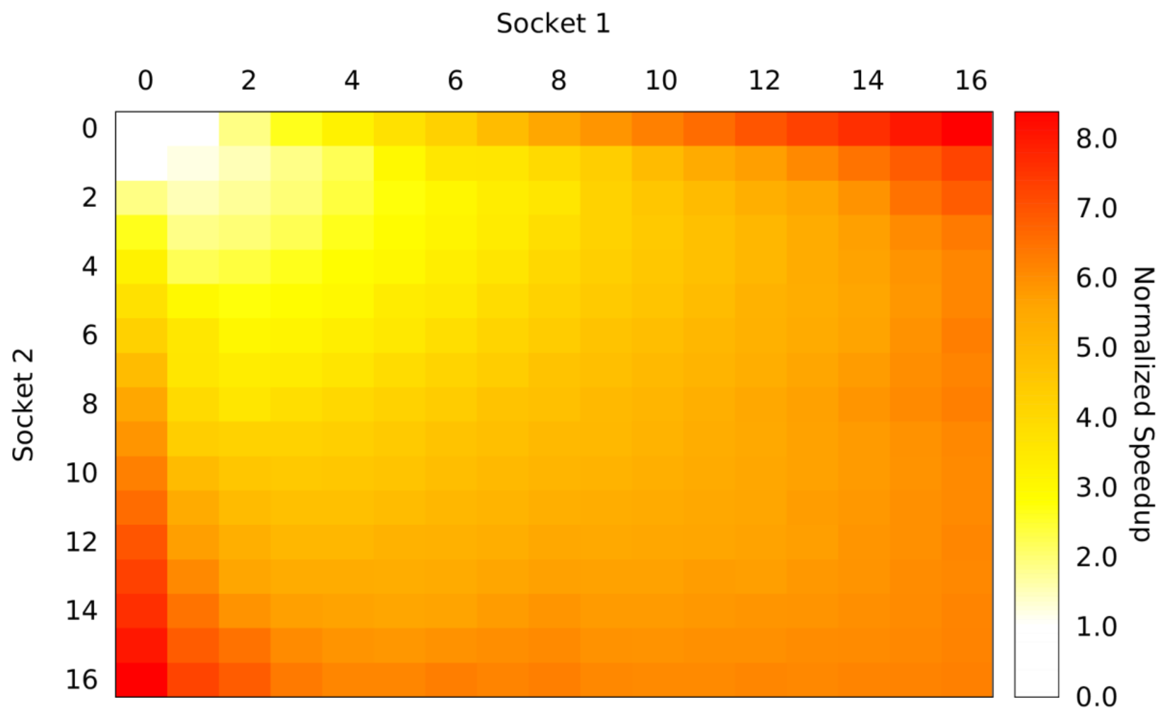
ORACLE

Example behaviour patterns

Parallel radix join optimized



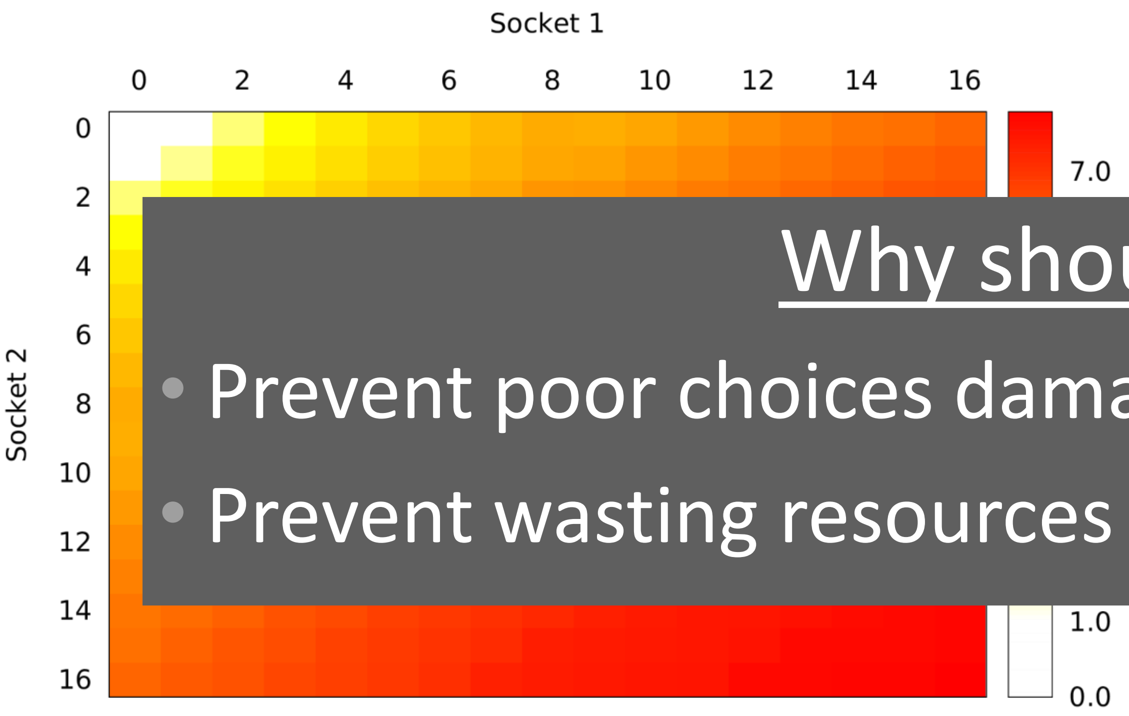
Molecular dynamics simulation



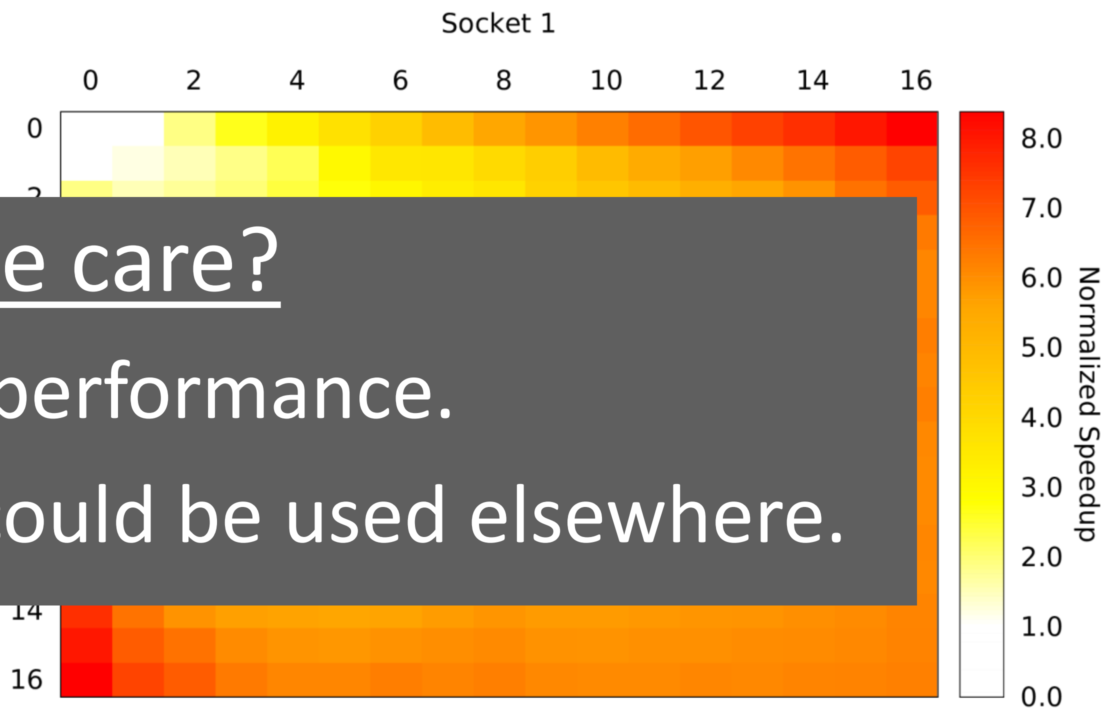
ORACLE

Example behaviour patterns

Parallel radix join optimized



Molecular dynamics simulation



Why should we care?

- Prevent poor choices damaging performance.
- Prevent wasting resources that could be used elsewhere.

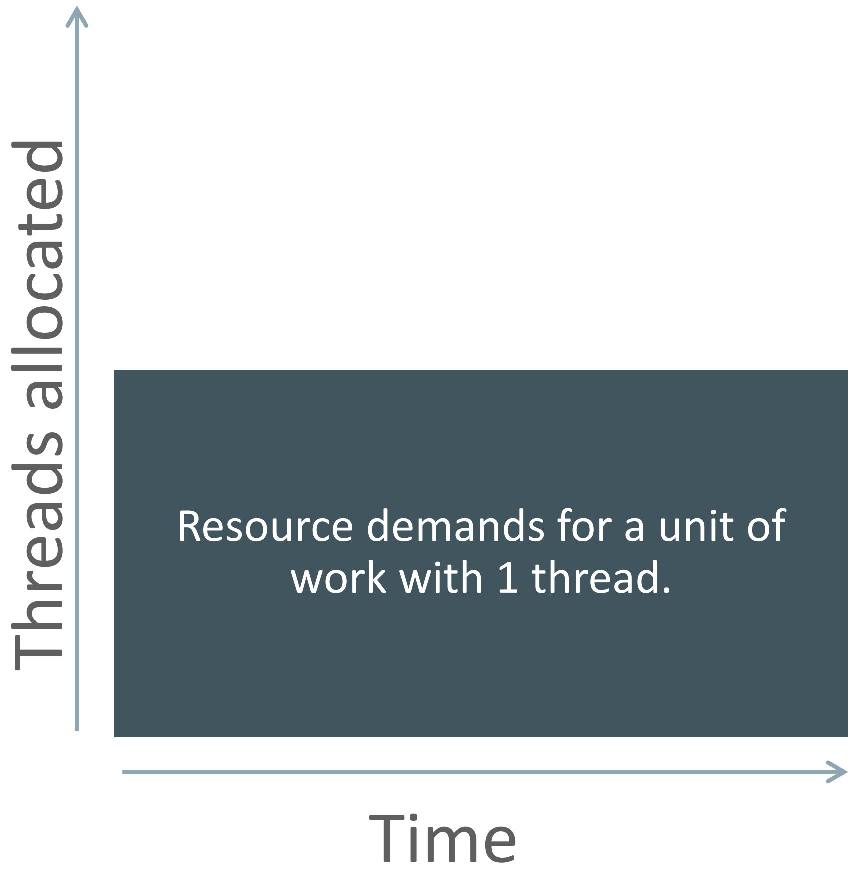
Overview

Key idea 1

Our workloads perform a roughly constant amount of work

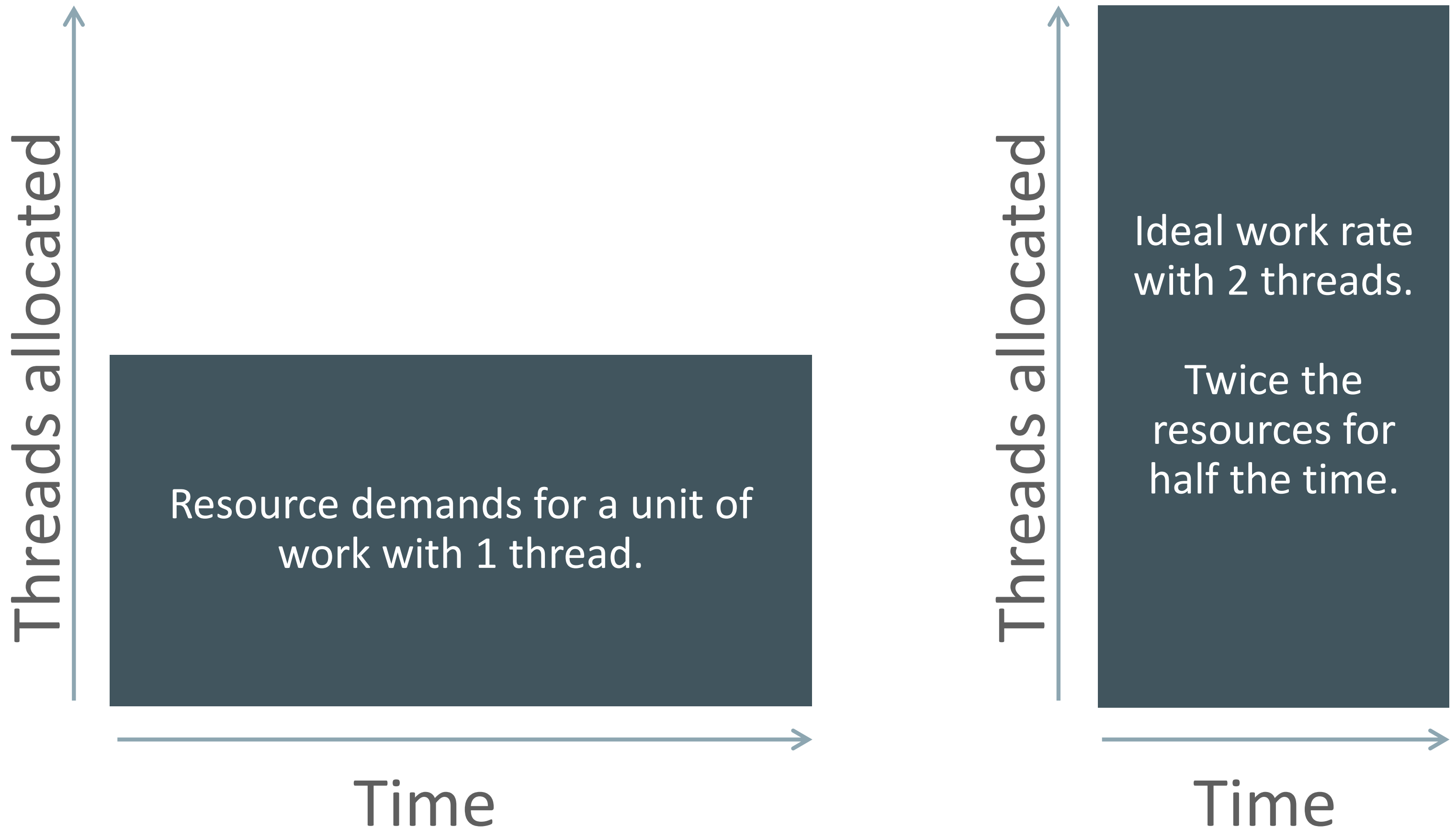
Key idea 1

Our workloads perform a roughly constant amount of work



Key idea 1

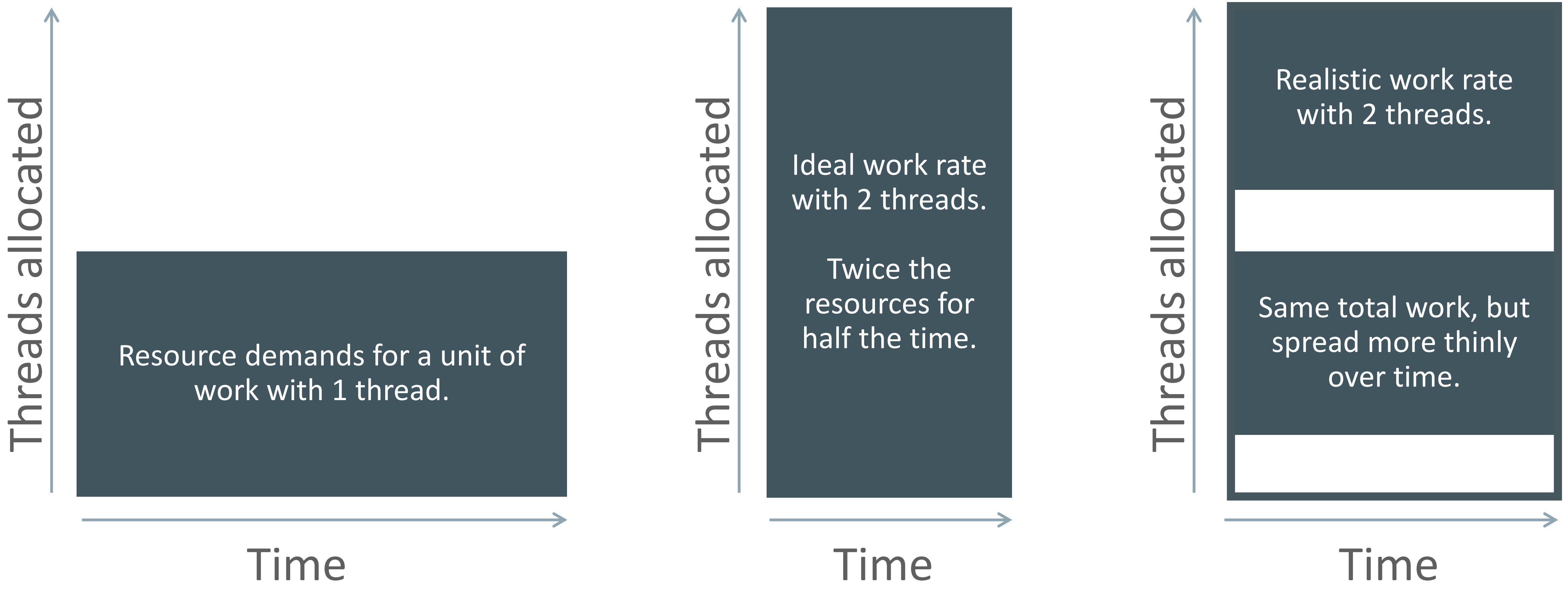
Our workloads perform a roughly constant amount of work



ORACLE

Key idea 1

Our workloads perform a roughly constant amount of work



ORACLE

Key idea 2

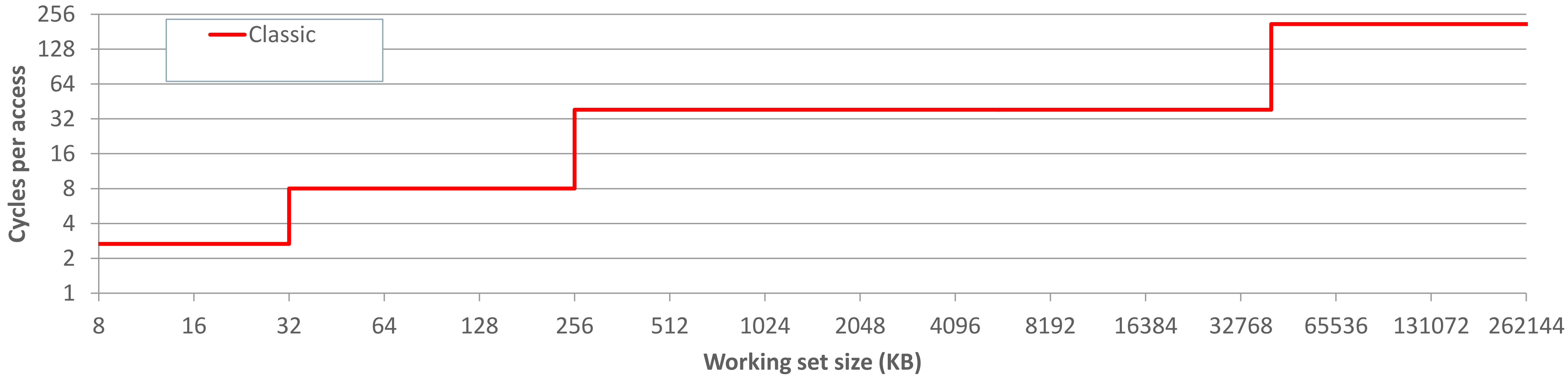
Hardware is getting ever more complicated, but looks simpler

- This makes it harder to model using conventional techniques, but removes much of the need to model behaviour in detail.

Key idea 2

Hardware is getting ever more complicated, but looks simpler

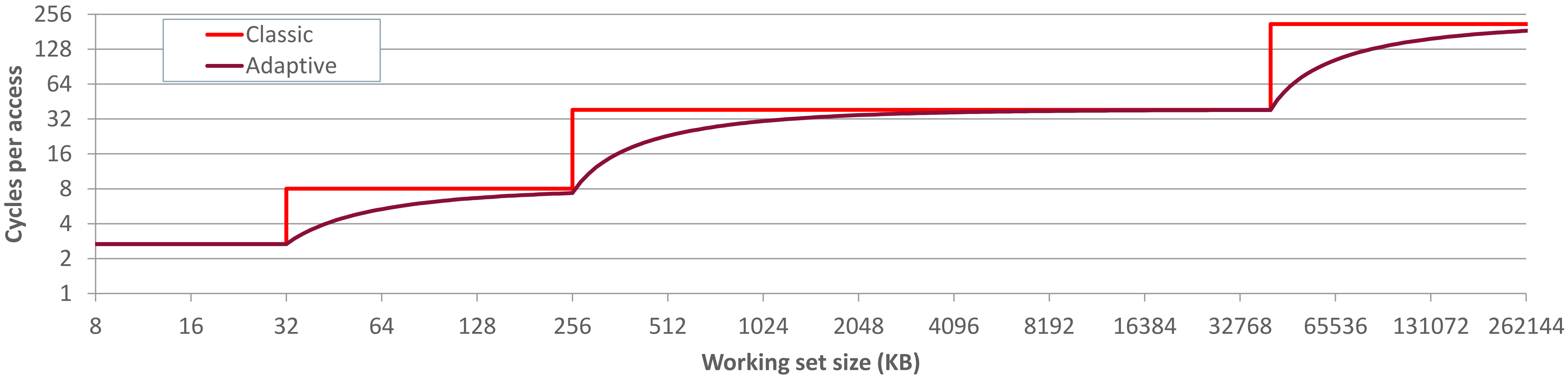
- This makes it harder to model using conventional techniques, but removes much of the need to model behaviour in detail.
- Example: Adaptive caches.



Key idea 2

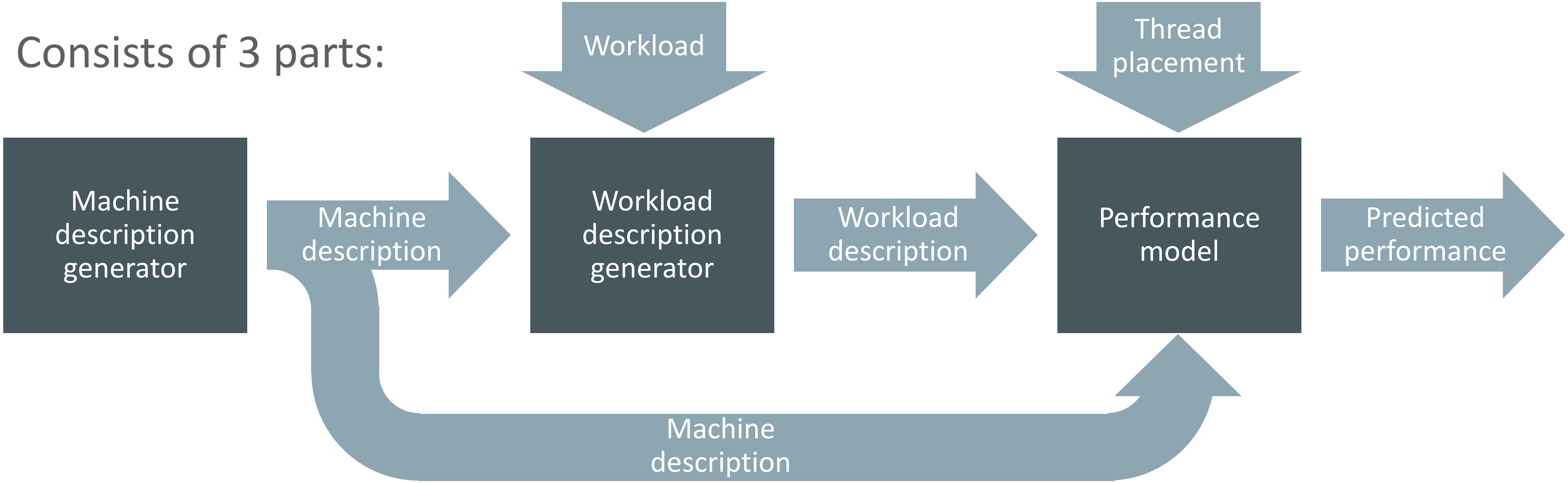
Hardware is getting ever more complicated, but looks simpler

- This makes it harder to model using conventional techniques, but removes much of the need to model behaviour in detail.
- Example: Adaptive caches.



Pandia: **predicting the performance of in-memory workloads**

Consists of 3 parts:

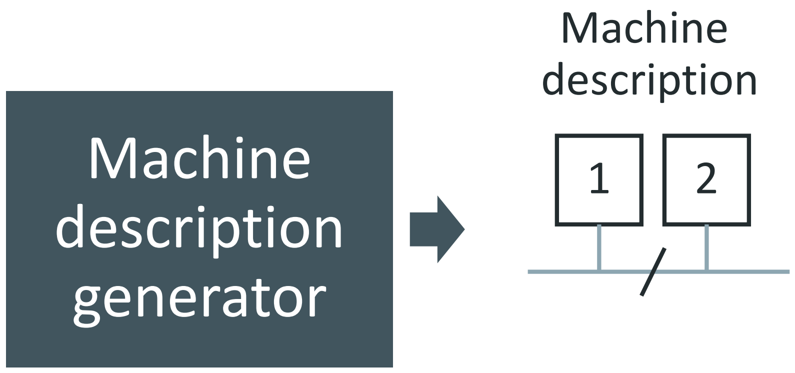


Machine description

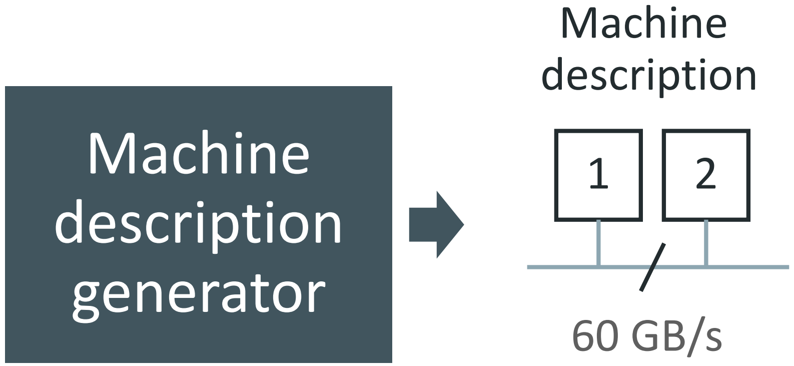
Machine description generator

Machine description

- Query the OS/processor for CPU count, core count, cache sizes, …

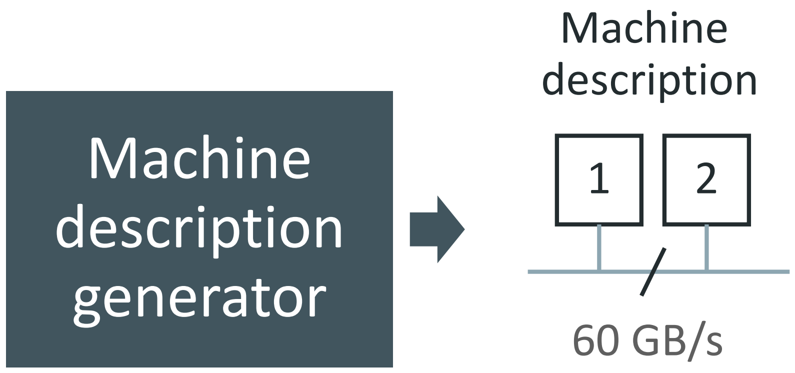


Machine description



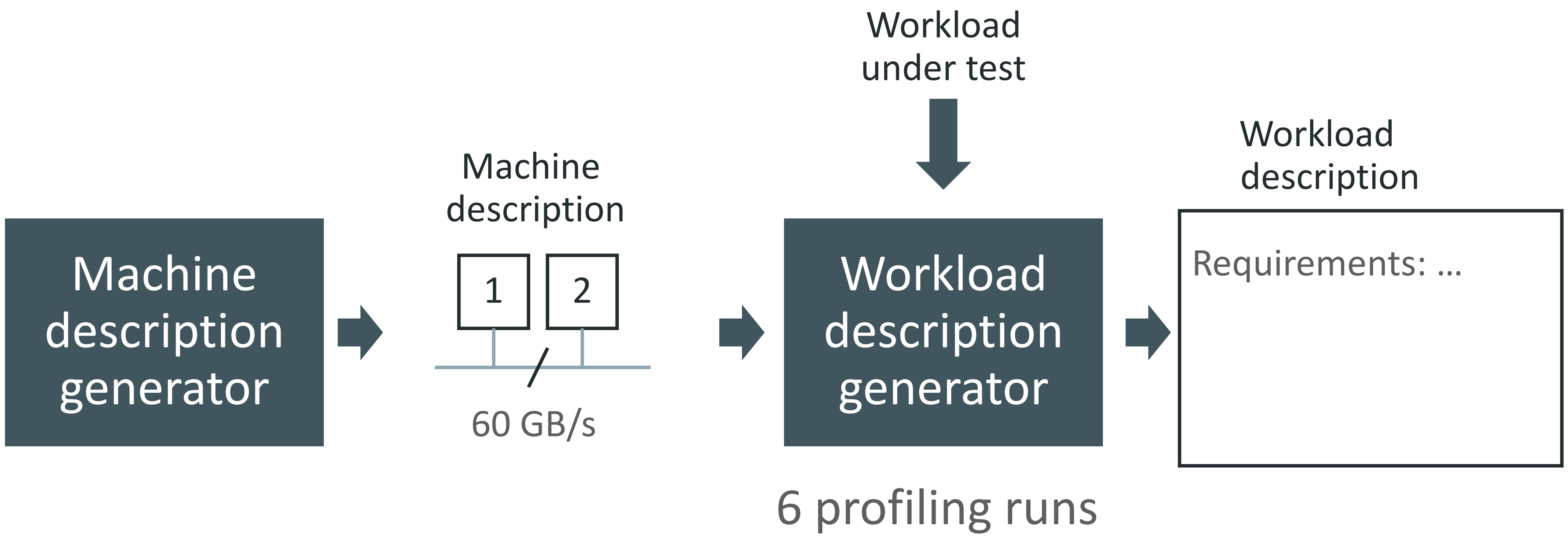
- Query the OS/processor for CPU count, core count, cache sizes, …
- Measure synthetic stress applications for:
  - Latency
  - Bandwidth
  - Execution rate (normalized IPC)
  - …

Machine description



- Query the OS/processor for CPU count, core count, cache sizes, …
- Measure synthetic stress applications for:
  - Latency
  - Bandwidth
  - Execution rate (normalized IPC)
  - …
- Detailed statistics are gained from performance counters

Workload description



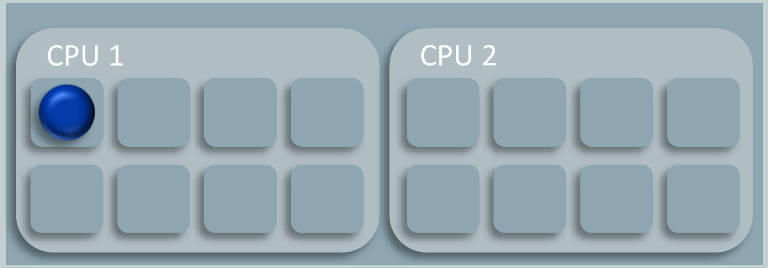
Workload description: per-thread requirements

- Characteristics that reflect the requirements of an individual thread:
  - Instruction execution rate (normalized instructions per cycle – IPC)
  - Memory bandwidth
  - Inter cache bandwidth
  - …
- Measured while running the application
  - Run the application with a minimal thread count
  - Record statistics using performance counters

ORACLE

Workload description: runs

Run 1: Single Thread



Workload description: parallelism characteristics

- Characteristics of the interactions between the threads
- These all reflect synchronization either at the hardware level or within the application

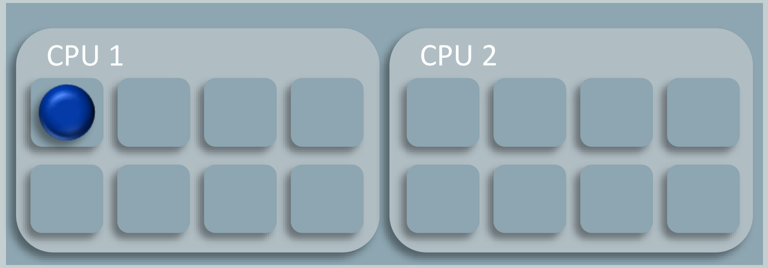
Workload description: parallelism characteristics

- Characteristics of the interactions between the threads
- These all reflect synchronization either at the hardware level or within the application.

Parallelism The percentage of the executed code that is parallel	

Workload description: runs

Run 1: single thread



Run 2: n threads, single socket

Workload description: parallelism characteristics

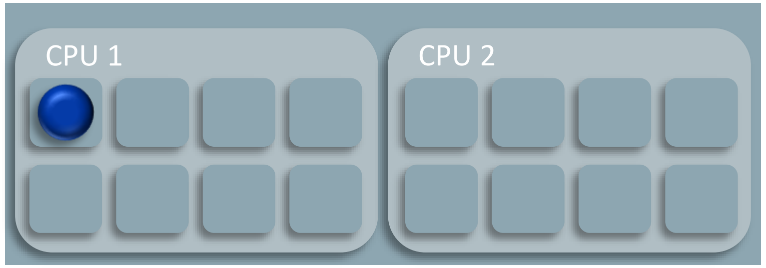
- Characteristics of the interactions between the threads
- These all reflect synchronization either at the hardware level or within the application

Parallelism The percentage of the executed code that is parallel	**Communication slowdown** Slowdown due to latency of thread communications between sockets, nodes, …

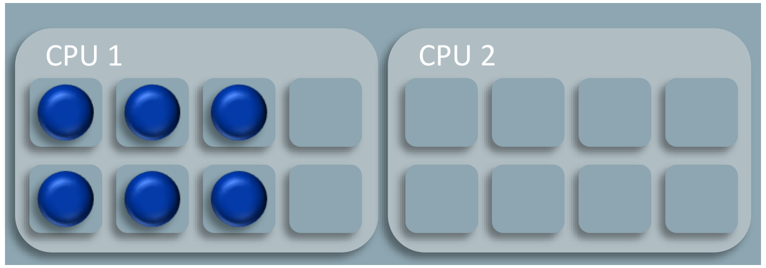
ORACLE

Workload description: runs

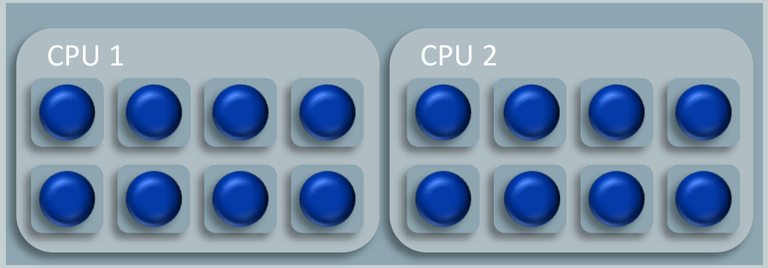
Run 1: single thread



Run 2: n threads, single socket



Run 3: symmetric, multi-socket



Workload description: parallelism characteristics

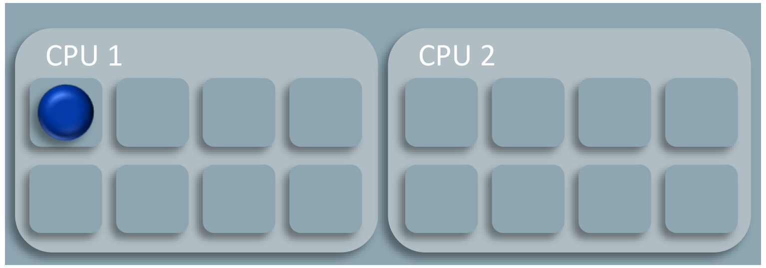
- Characteristics of the interactions between the threads
- These all reflect synchronization either at the hardware level or within the application

Parallelism The percentage of the executed code that is parallel	**Communication slowdown** Slowdown due to latency of thread communications between sockets, nodes, …
Thread interlocking Are threads independent?	

ORACLE

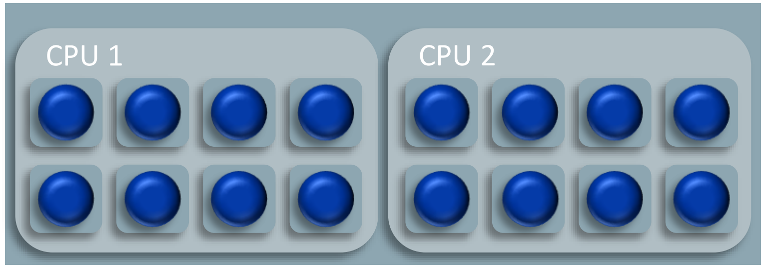
Workload description: runs

Run 1: single thread

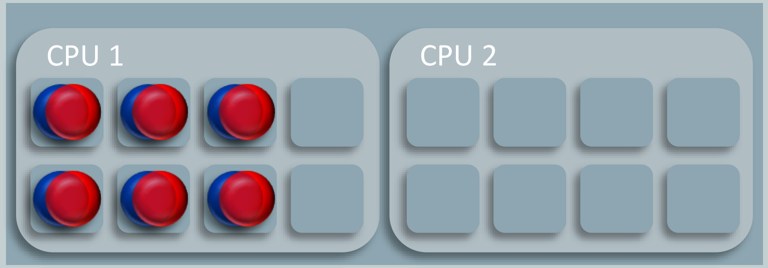


Run 2: n threads, single socket

Run 3: symmetric, multi-socket



Run 4: n threads, n slowed



Run 5: n threads, 1 slowed

ORACLE

Workload description: parallelism characteristics

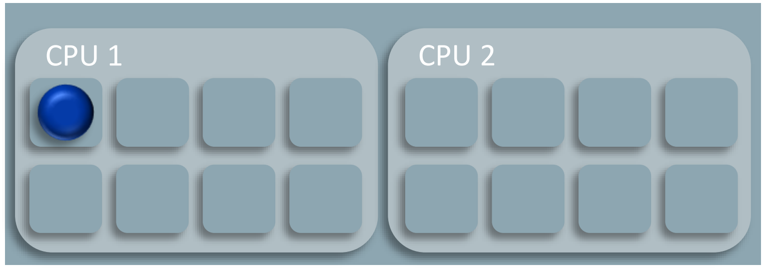
- Characteristics of the interactions between the threads
- These all reflect synchronization either at the hardware level or within the application

Parallelism The percentage of the executed code that is parallel	**Communication slowdown** Slowdown due to latency of thread communications between sockets, nodes, …
Thread interlocking Are threads independent?	**Coincident resource demands** Do threads have synchronized mode changes (e.g., intensive reads followed by intensive CPU)

ORACLE

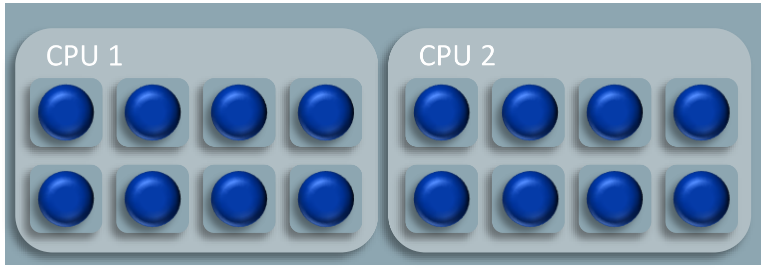
Workload description: runs

Run 1: Single thread

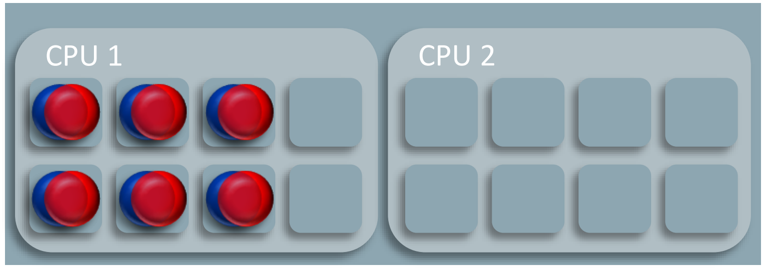


Run 2: n threads, single socket

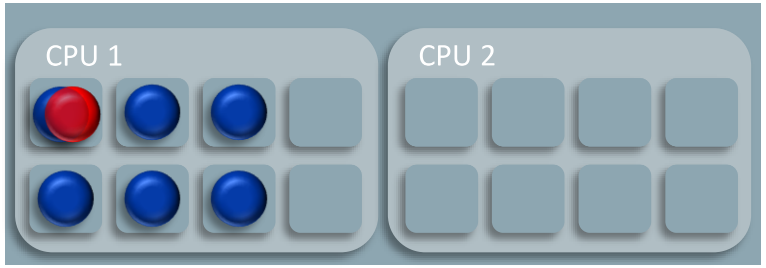
Run 3: symmetric, multi-socket



Run 4: n threads, n slowed

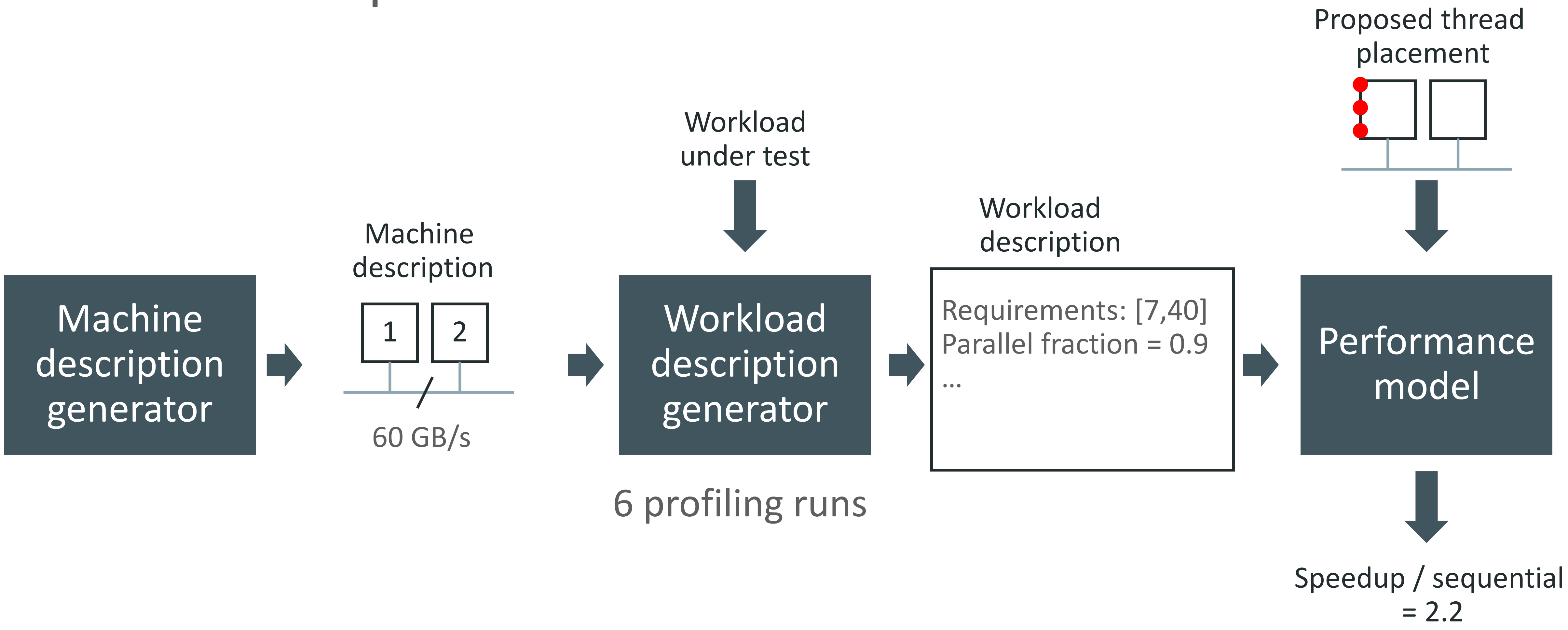


Run 5: n threads, 1 slowed



Run 6: n threads, hyper-threaded

Performance prediction



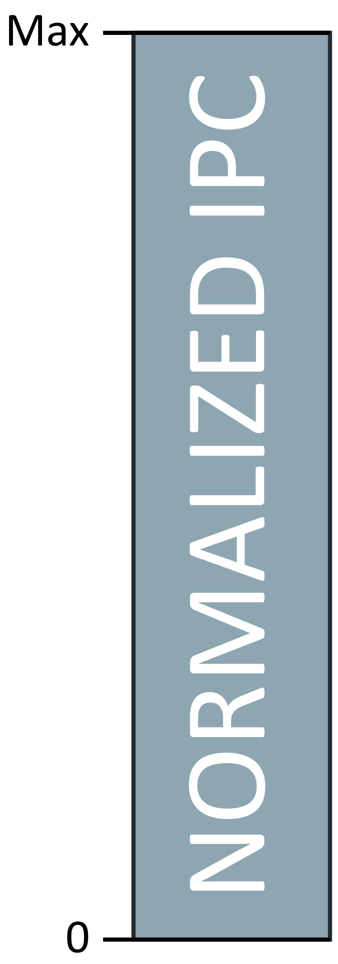
Performance model

$$\text{Predicted speedup} = \frac{\text{Estimated speedup with thread count (Amdahl's law)}}{\text{Estimated slowdown due to resource contention}}$$

ORACLE

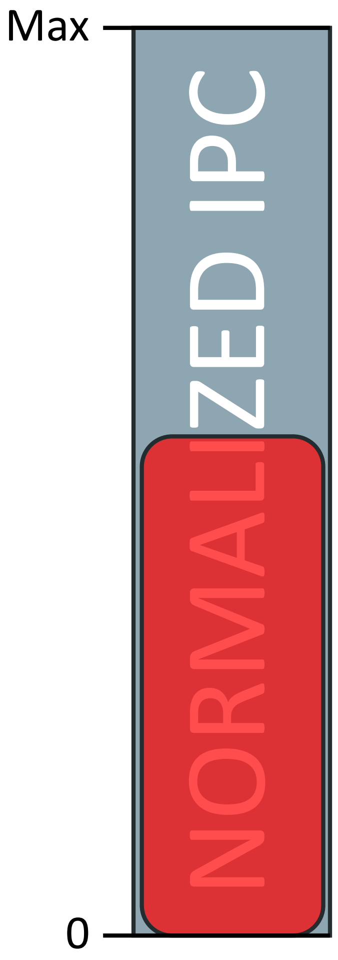
Estimating thread slowdown

- A machine is a set of components each with a set of resources
- Over subscribing any of these resources will produce a slowdown
- The nature of the slowdown will vary with the resource



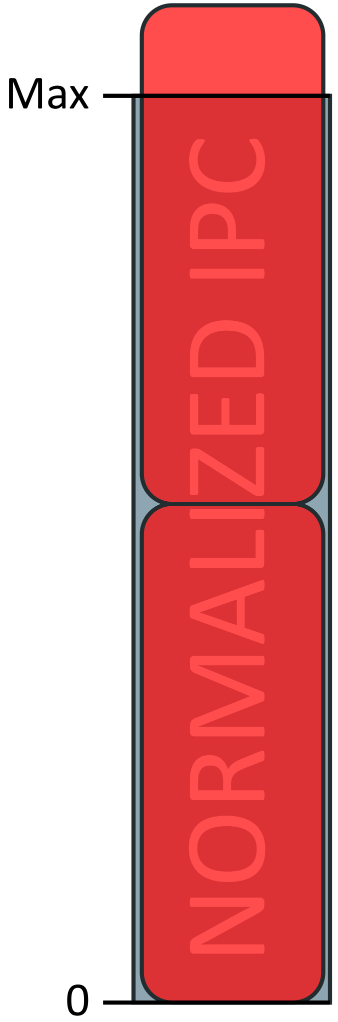
Estimating thread slowdown

- A machine is a set of components each with a set of resources
- Over subscribing any of these resources will produce a slowdown
- The nature of the slowdown will vary with the resource



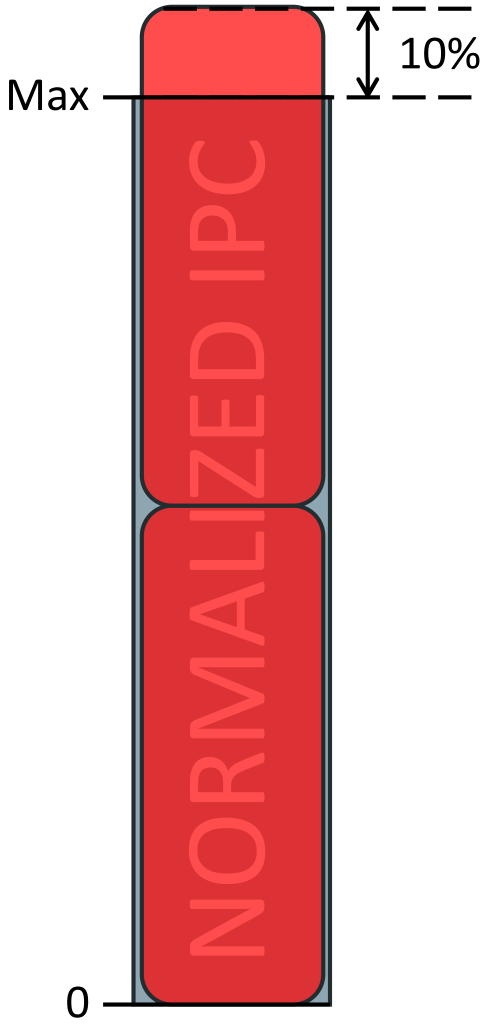
Estimating thread slowdown

- A machine is a set of components each with a set of resources
- Over subscribing any of these resources will produce a slowdown
- The nature of the slowdown will vary with the resource



Estimating thread slowdown

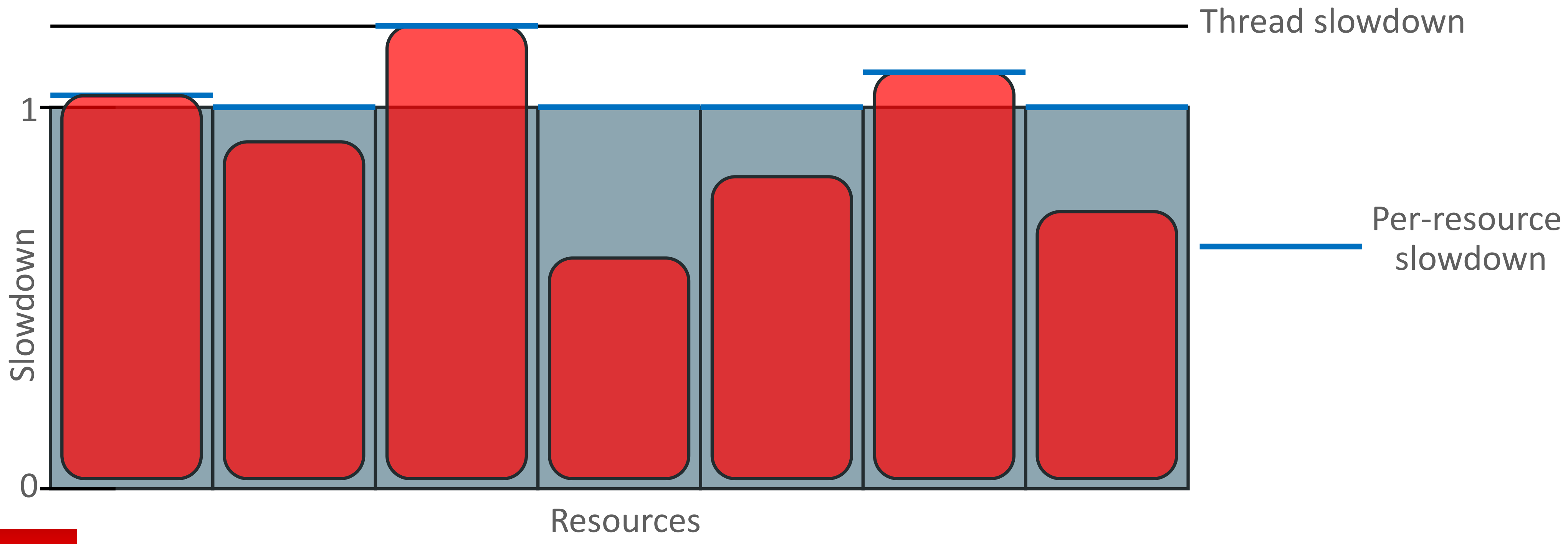
- A machine is a set of components each with a set of resources
- Over subscribing any of these resources will produce a slowdown
- The nature of the slowdown will vary with the resource



$$Slowdown = \frac{1.1}{1} = 1.1$$

Estimating thread slowdown

- Each thread has a slowdown calculated for all the resources it uses
- Each thread's slowdown is the maximum of the slowdowns it encounters



Workload description: parallelism characteristics

- Characteristics of the interactions between the threads
- These all reflect synchronization either at the hardware level or within the application

Parallelism The percentage of the executed code that is parallel	**Communication slowdown** Slowdown due to latency of thread communications between sockets, nodes, …
Thread interlocking Are threads independent?	**Coincident resource demands** Do threads have synchronized mode changes (e.g., intensive reads followed by intensive CPU)

ORACLE

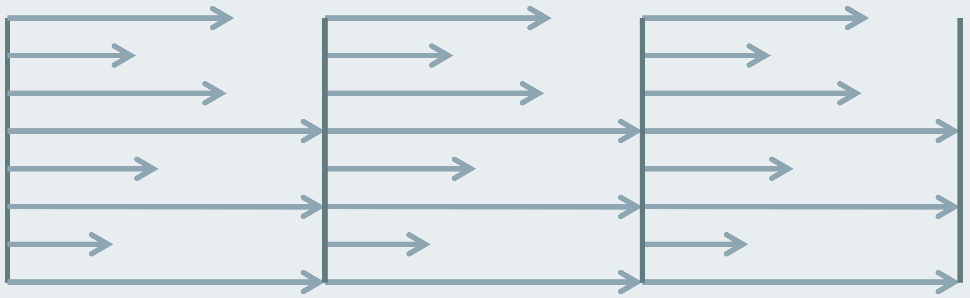
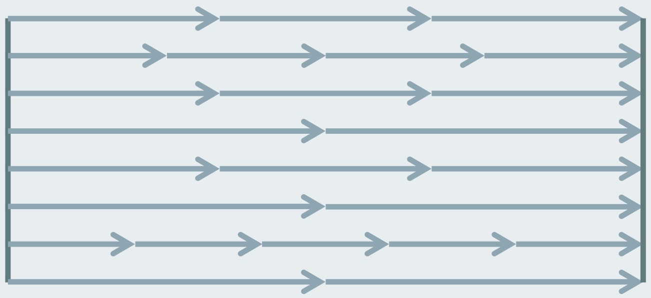
Workload description: parallelism characteristics

- Characteristics of the interactions between the threads
- These all reflect synchronization either at the hardware level or within the application

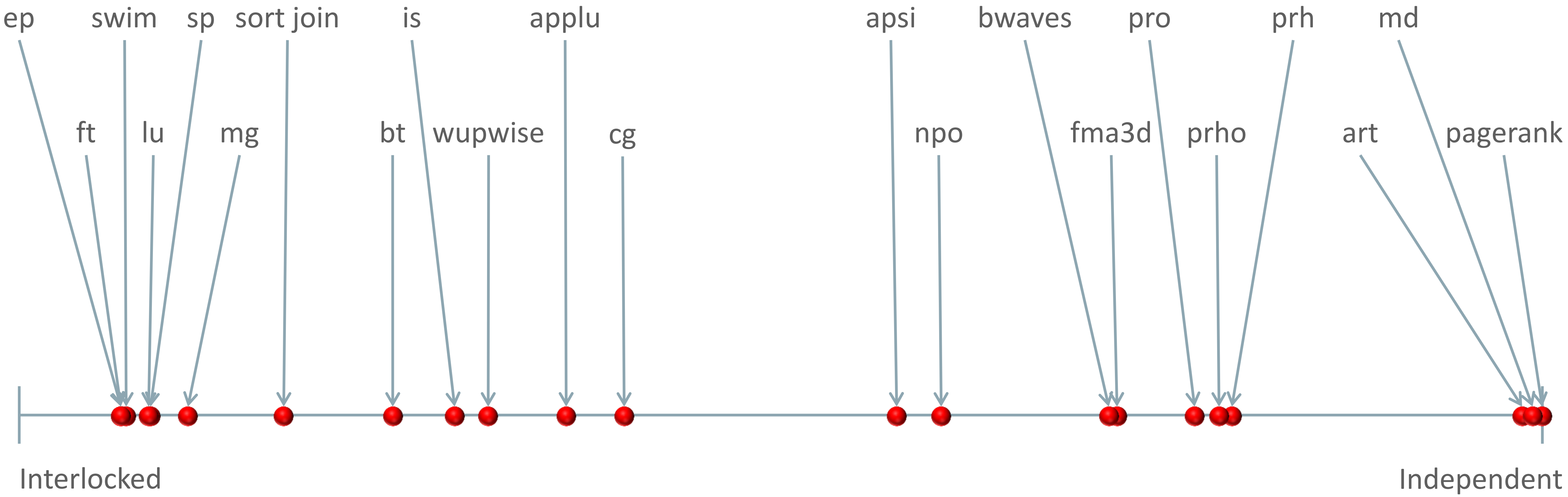
Parallelism The percentage of the executed code that is parallel	**Communication slowdown** Slowdown due to latency of thread communications between sockets, nodes, …
Thread interlocking Are threads independent?	**Coincident resource demands** Do threads have synchronized mode changes (e.g., intensive reads followed by intensive CPU)

Thread interlocking

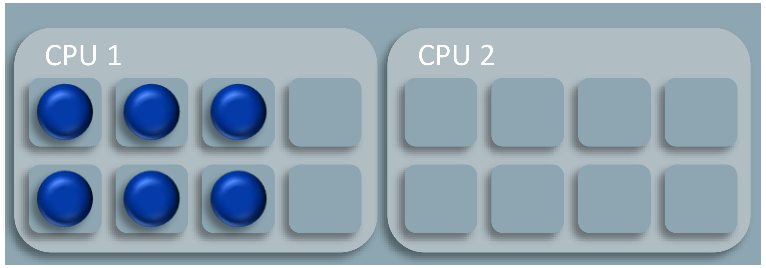
- Describes how to combine thread slowdowns:

Interlocked threads	Independent threads
Each thread does equal work	Each thread works for the same time
	
Return the lowest rate of work	Return the average rate of work

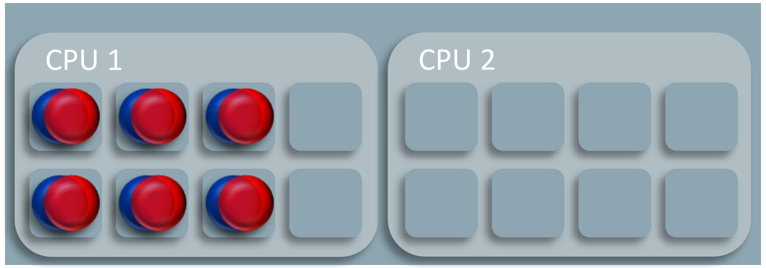
Thread interlocking



Thread interlocking

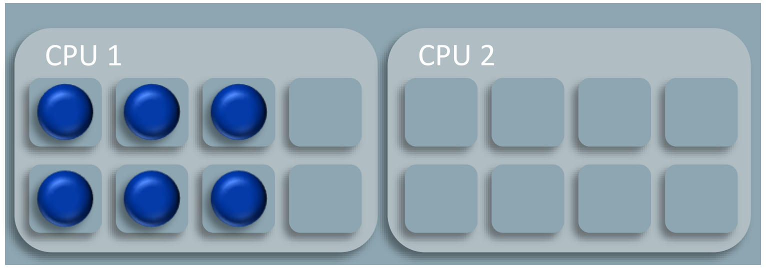


Run 2: Normal execution

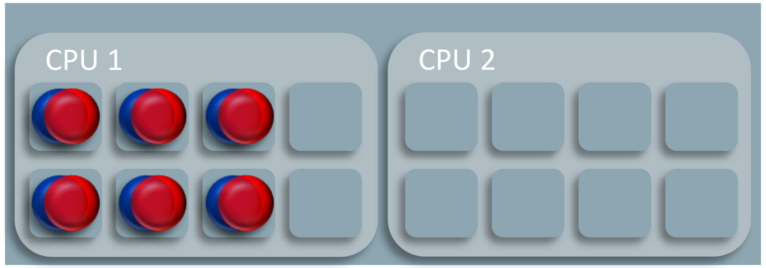


Run 4: All threads slowed –
this provides the slowdown per thread

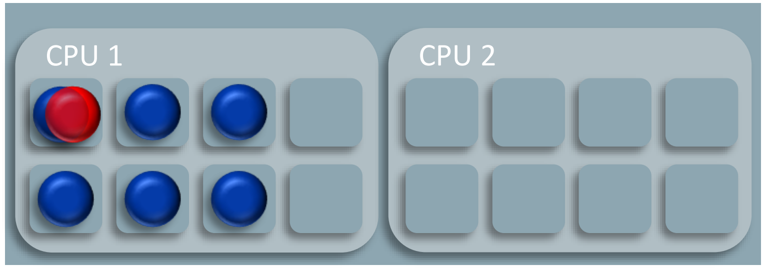
Thread interlocking



Run 2: Normal execution



Run 4: All threads slowed –
this provides the slowdown per thread



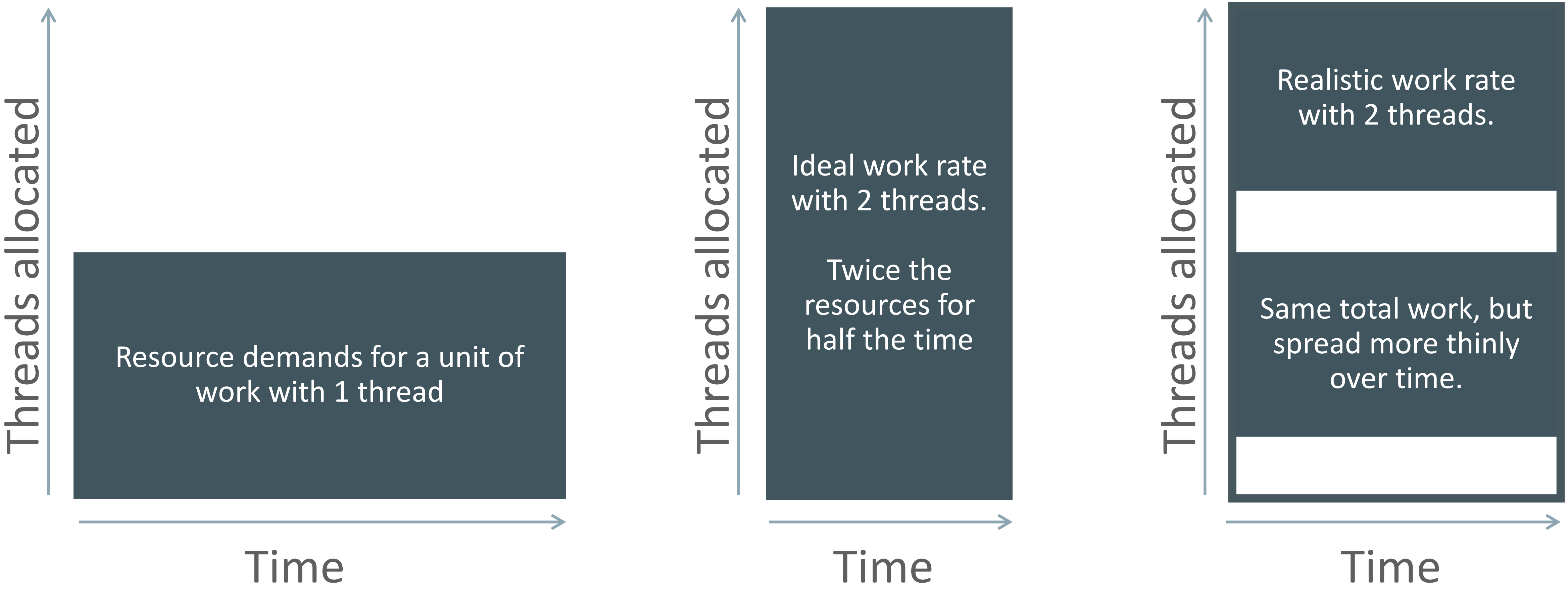
Run 5: One thread slowed –
this run’s time is interpolated between the extremes

Performance model

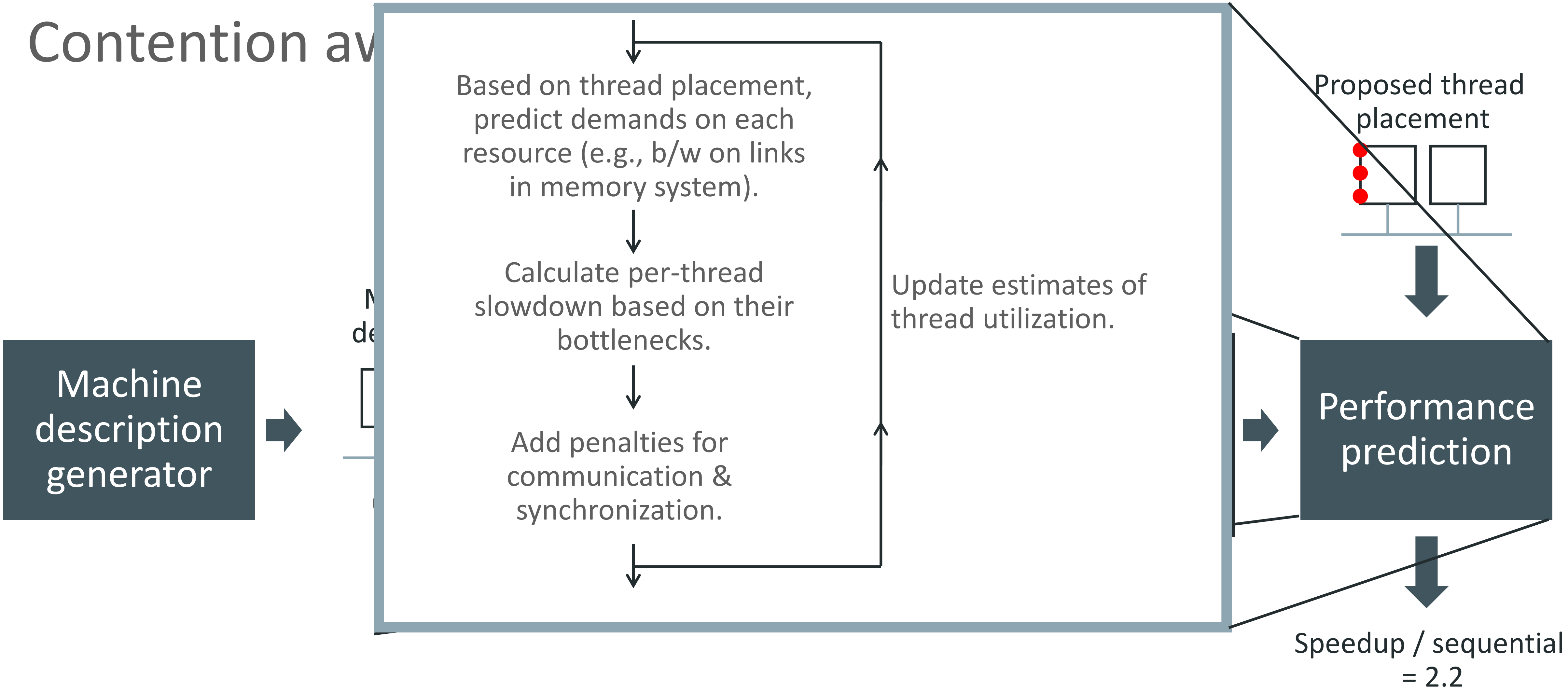
$$\text{Predicted speedup} = \frac{\text{Estimated speedup with thread count (Amdahl's law)}}{\text{Estimated slowdown due to resource contention}}$$

Thread utilization

Workloads perform a roughly constant amount of work.



Contention av



Overview

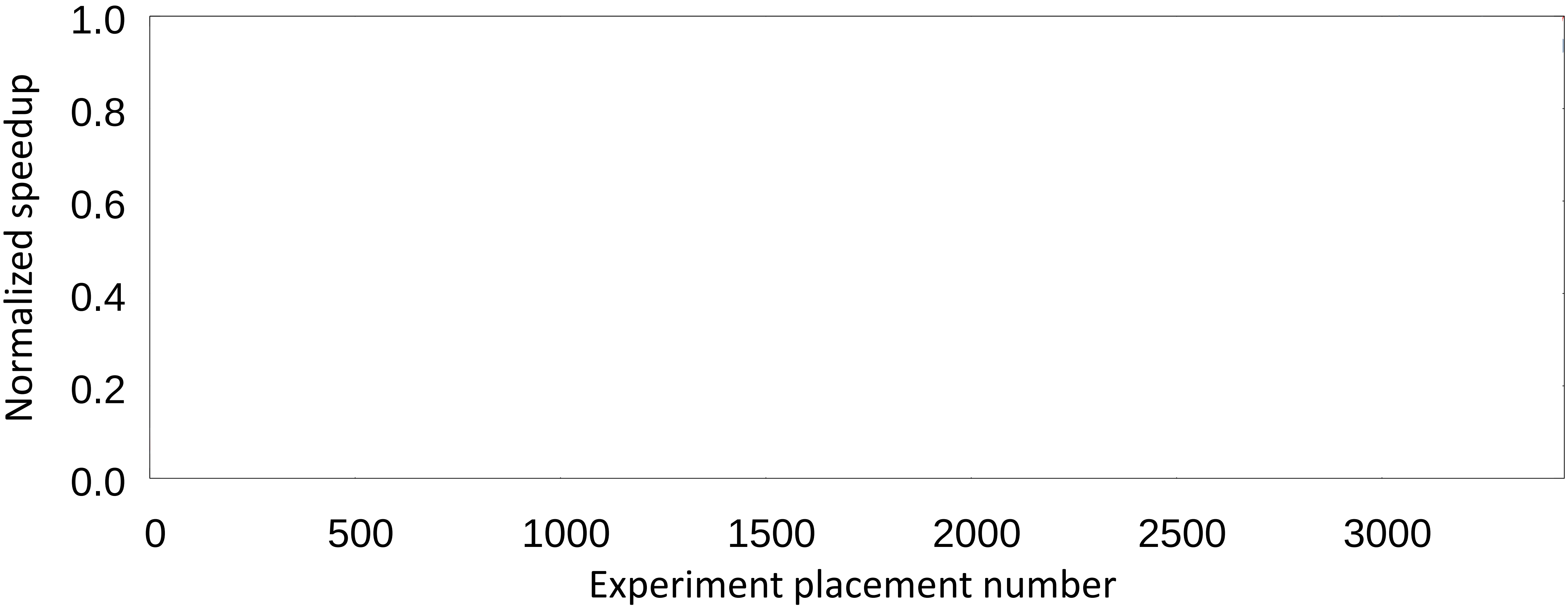
1. What is the problem?
2. How does Pandia work?
3. How well does Pandia work?
4. Conclusions

Evaluation

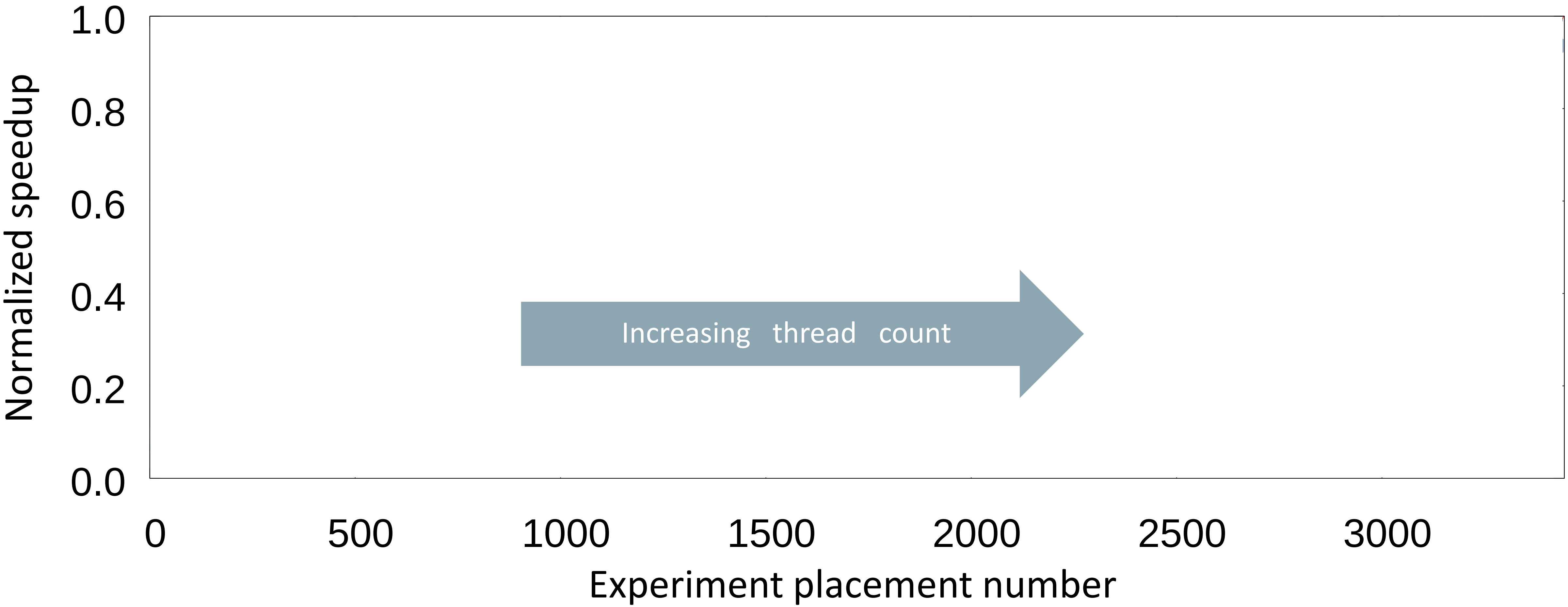
- Evaluated on a range of systems:
  - 2 socket Haswell, Ivy-bridge, Sandy-bridge
  - 4 socket Westmere
- Developed using 4 benchmarks (BT, CG, IS, MD), 18 additional benchmarks used in evaluation
- Profiles contain 9 parameters, but each test generates 1000s of data points
- Model features are tied to observable hardware and program features, not to features of the dataset
- Test the portability of workload descriptions between machines

ORACLE

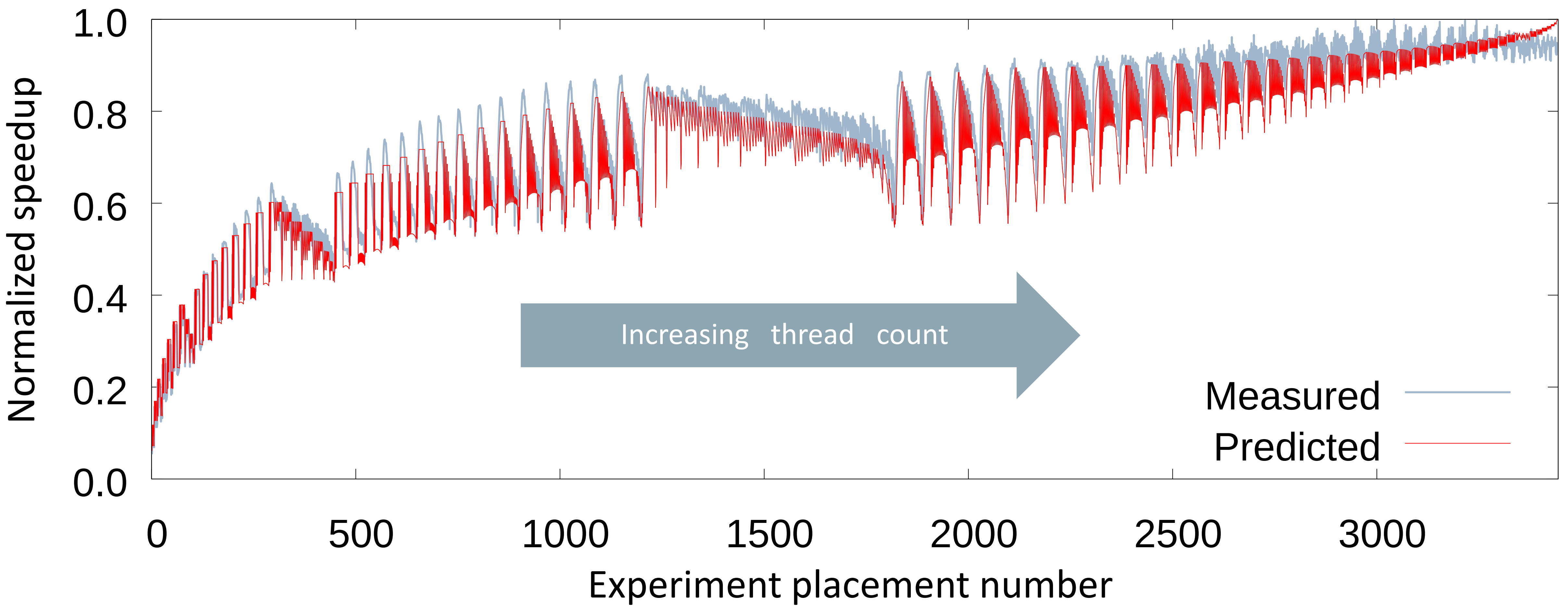
Predicted v measured performance



Predicted v measured performance

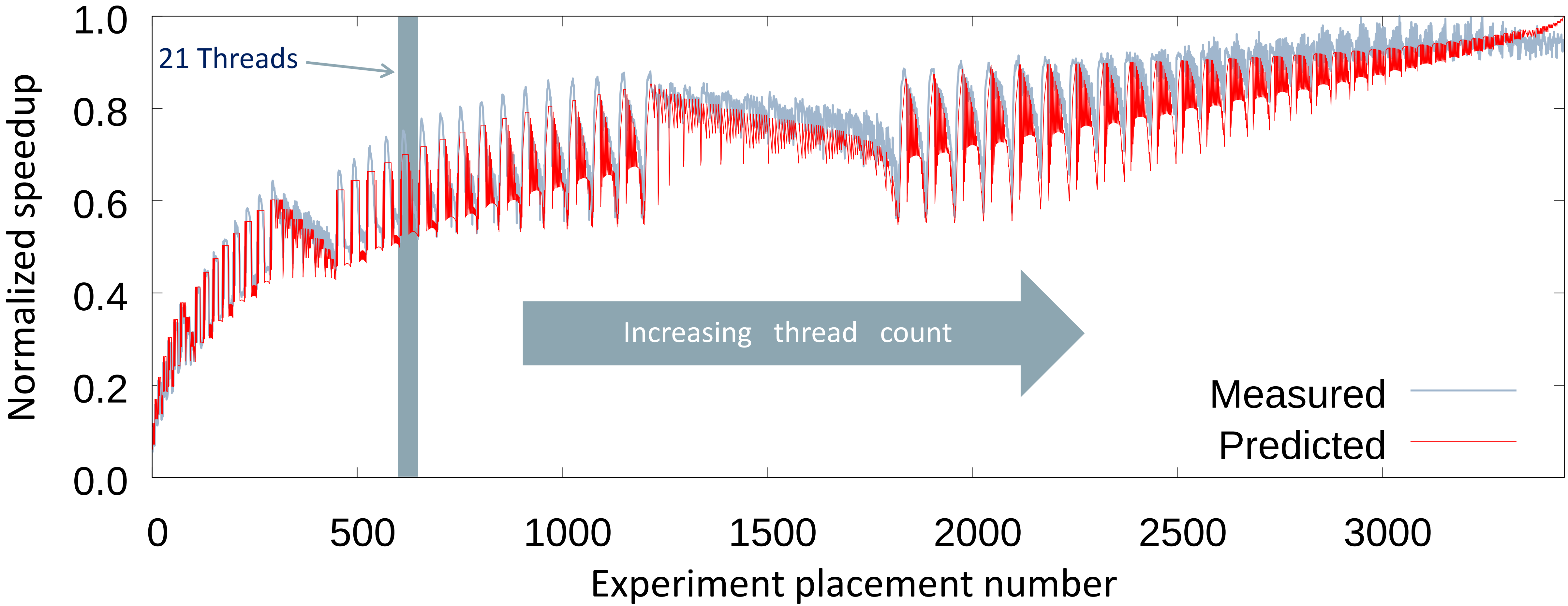


Predicted v measured performance (Pagerank)

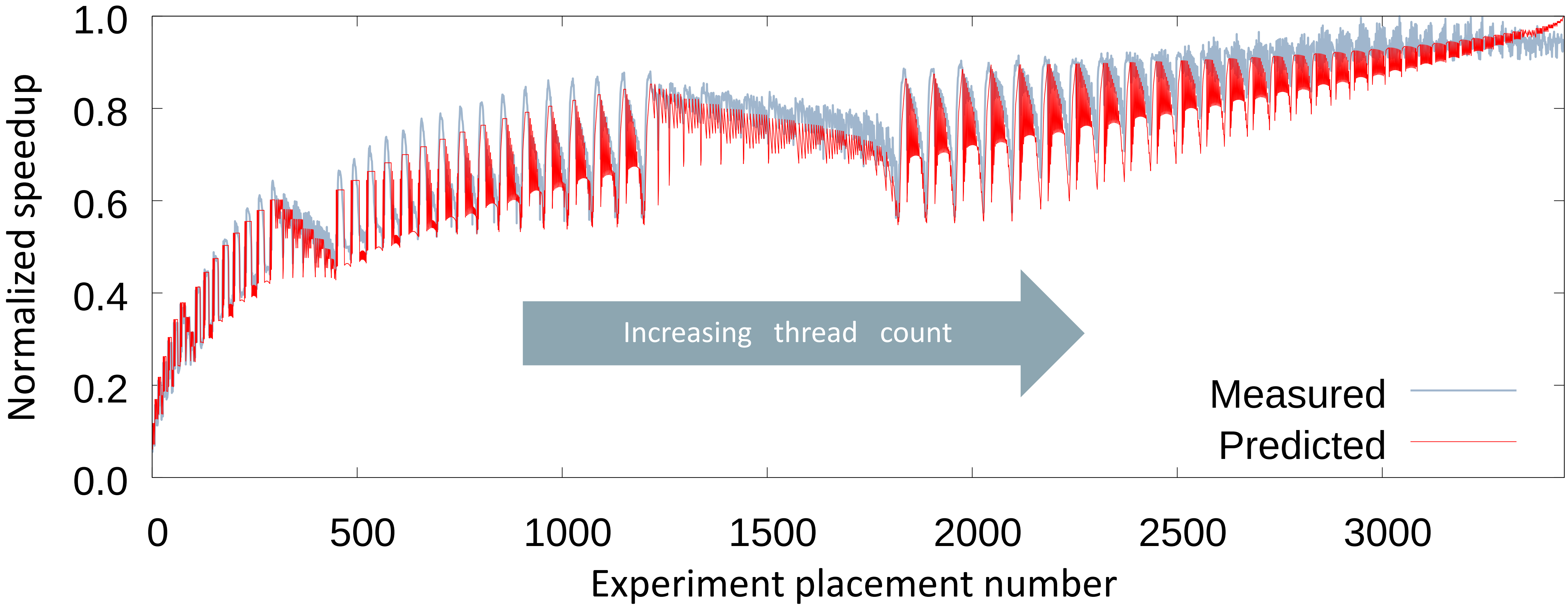


ORACLE

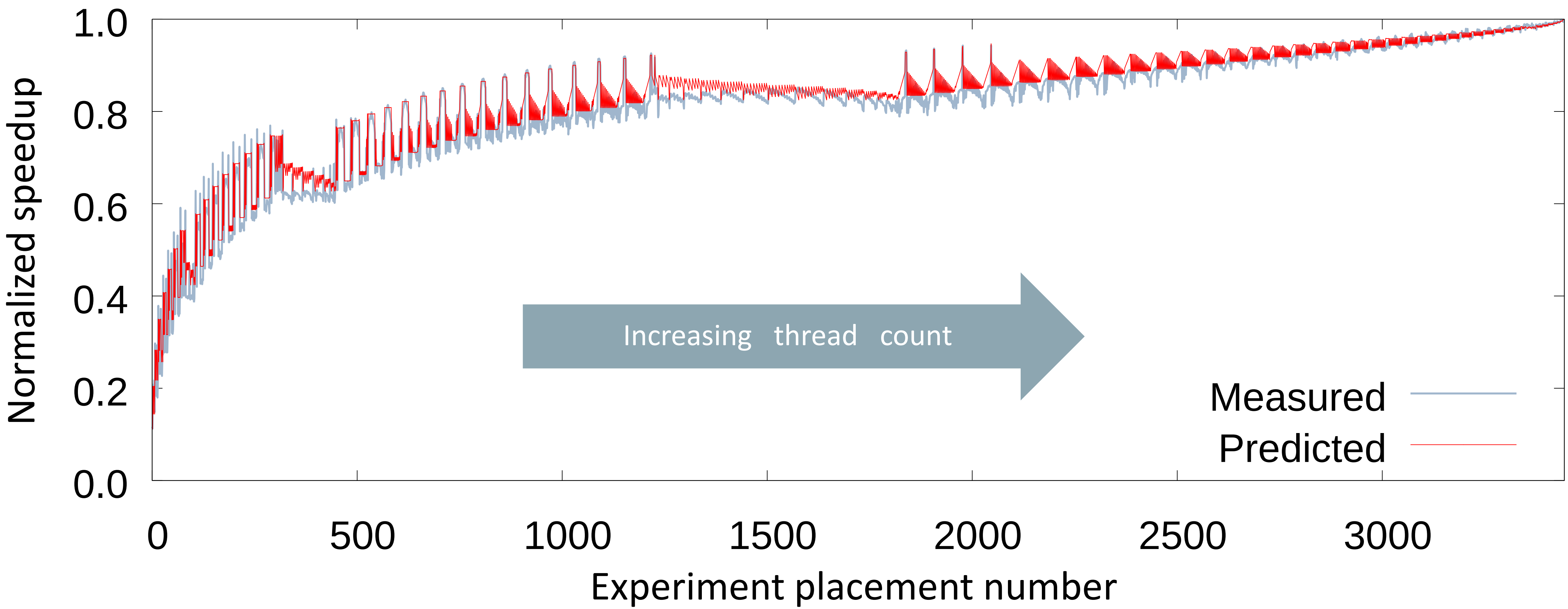
Predicted v measured performance (Pagerank)



Predicted v measured performance (Pagerank)

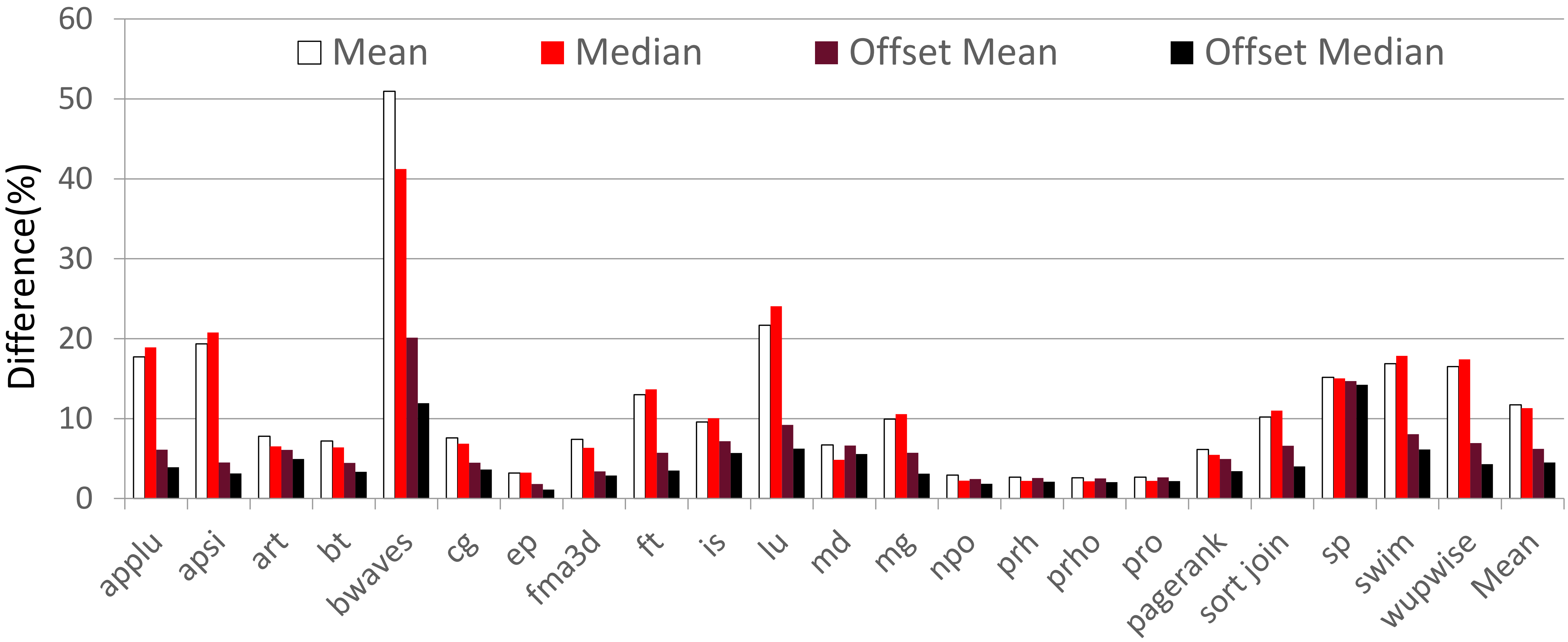


Predicted v measured performance (Database hash join)



ORACLE

Average error



ORACLE

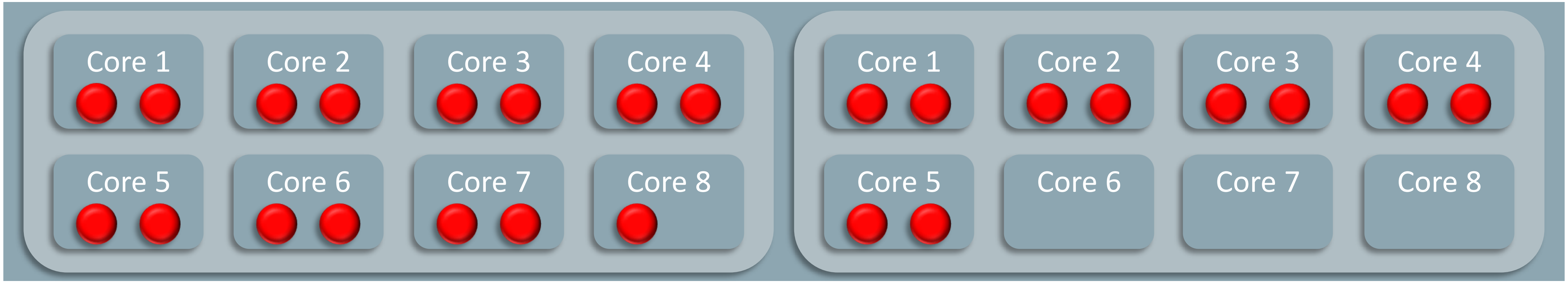
Best vs predicted best placement

- Larger thread counts are more accurate

Machine	Mean	Median
2 socket Sandy-bridge	0.77%	0.00%
2 socket Ivy-bridge	0.29%	0.00%
2 socket Haswell	2.78%	1.05%

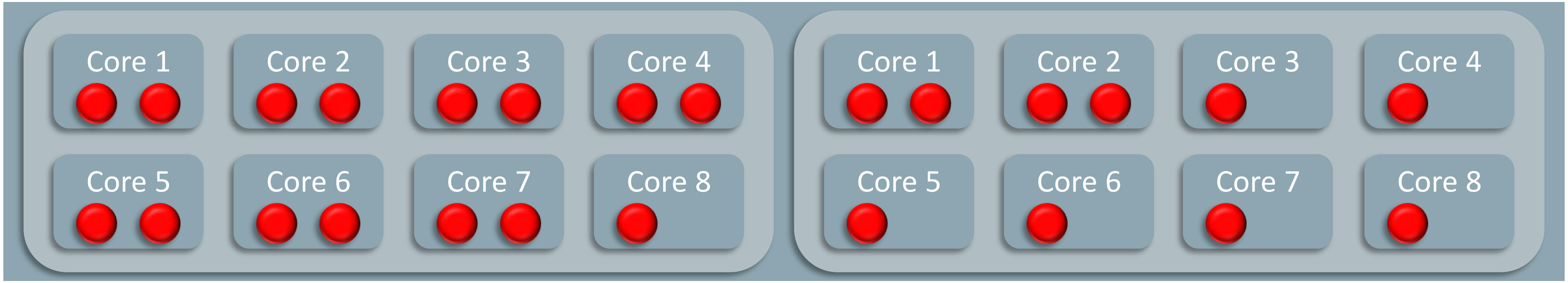
Measured-best vs predicted-best placement (BT)

Measured-best



Predicted-best

Performance loss 0.36%



Overview

ORACLE

Conclusions

What we expected

Modeling synchronization | Cache effects | NUMA effects

What we found

Conclusions

What we expected

Workloads can involve complex mixes of barriers, locks, atomics, etc.

Modeling synchronization

Cache effects

NUMA effects

What we found

Averaging the effects with a simple model based on Amdahl's law was sufficient.

ORACLE

Conclusions

What we expected

	Modeling synchronization	Cache effects	NUMA effects
What we expected	Workloads can involve complex mixes of barriers, locks, atomics, etc.	Source of complexity, and significant earlier exploration.	
What we found	Averaging the effects with a simple model based on Amdahl's law was sufficient.	Simple bandwidth based model suffices. More complex h/w mitigates performance cliffs.	

What we found

ORACLE

Conclusions

What we expected

Modeling synchronization	Cache effects	NUMA effects
Workloads can involve complex mixes of barriers, locks, atomics, etc.	Source of complexity, and significant earlier exploration.	How to distinguish impact of local vs remote memory accesses?
Averaging the effects with a simple model based on Amdahl's law was sufficient.	Simple bandwidth based model suffices. More complex h/w mitigates performance cliffs.	Uniformity across the workload helps again: measure performance in aggregate.

What we found

ORACLE

Conclusions

- Modern hardware avoids many of the pathological performance cases
- Simple models can be good enough to make meaningful decisions
- Predictions include resource predictions
- Best placements not always found by exploring scatter and pack placements
- State exploration will only get more complex when considering multiple workloads, so technique like Pandia are needed

For information about Pandia or roles in Oracle Labs please get in touch – daniel.goodman@oracle.com

ORACLE

Integrated Cloud

Applications & Platform Services

ORACLE